

1213.43347X00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): NAKAMURA, et al
Serial No.:
Filed: December 17, 2003
Title: DISTRIBUTED FILE SYSTEM
Group:

LETTER CLAIMING RIGHT OF PRIORITY

Mail Stop Patent Application
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

December 17, 2003

Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55, the applicant(s) hereby claim(s) the right of priority based on Japanese Patent Application No.(s) 2003-063569 filed March 10, 2003.

A certified copy of said Japanese Application is attached.

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP



Carl I. Brundidge
Registration No. 29,621

CIB/nac
Attachment
(703) 312-6600

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

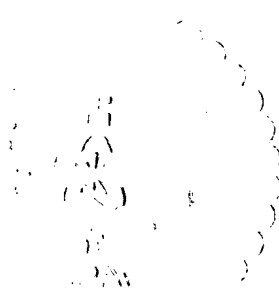
This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年 3月10日
Date of Application:

出願番号 特願2003-063569
Application Number:

[ST. 10/C]: [JP 2003-063569]

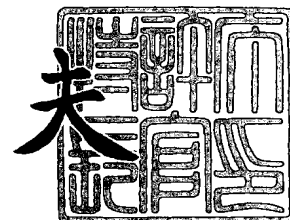
出願人 株式会社日立製作所
Applicant(s):



2003年 9月26日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫



出証番号 出証特2003-3079397

【書類名】 特許願

【整理番号】 GM0303014

【提出日】 平成15年 3月10日

【あて先】 特許庁長官殿

【国際特許分類】 G16F 12/00

【発明者】

【住所又は居所】 東京都国分寺市東恋ヶ窪一丁目 2 8 0 番地 株式会社日立製作所 中央研究所内

【氏名】 中村 隆喜

【発明者】

【住所又は居所】 東京都品川区東品川四丁目 1 2 番 7 号 日立ソフトウェアエンジニアリング株式会社内

【氏名】 小山 和成

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100075513

【弁理士】

【氏名又は名称】 後藤 政喜

【選任した代理人】

【識別番号】 100084537

【弁理士】

【氏名又は名称】 松田 嘉夫

【選任した代理人】

【識別番号】 100114236

【弁理士】

【氏名又は名称】 藤井 正弘

【手数料の表示】

【予納台帳番号】 019839

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0110326

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 分散ファイルシステム

【特許請求の範囲】

【請求項 1】

ファイルを保持するストレージと、前記ストレージに対してファイルの操作を行う複数のクライアントと、前記クライアントによるファイルの書き込み及び読み出しの操作の権限をトークンによって管理するサーバと、がネットワークによって接続されて構成される分散ファイルシステムにおいて、

前記サーバは、前記ファイルの書込操作権限を保持しているクライアントに対して該書込操作権限に関するトークンの返却を要求するトークン返却要求を送信するトークン返却要求手段を備え、

前記トークン返却要求手段は、該ファイルに関するトークンを要求しているクライアントの情報と、該クライアントが要求しているトークンの内容を示す情報とを含めてトークン返却要求を送信することを特徴とする分散ファイルシステム

。

【請求項 2】

前記クライアントは、前記ストレージから読み出したファイルのデータを保持する記憶部と、

前記トークン返却要求を受信すると、該ファイルに関するトークンを前記サーバに要求しているクライアントに対して、前記記憶部に保持している該トークンに関するファイルを送信するデータ出力手段と、を備えることを特徴とする請求項 1 に記載の分散ファイルシステム。

【請求項 3】

前記クライアントから前記トークンを前記サーバに要求しているクライアントに対して送信される該トークンに関するファイルは、前記ストレージに最新情報が反映されていないものであることを特徴とする請求項 2 に記載の分散ファイルシステム。

【請求項 4】

前記トークンには、前記ファイルの領域に関連づけられており、

前記データ出力手段は、

前記ファイルのうち、前記トークンによって関連づけられた領域のデータを、前記トークンを前記サーバに要求しているクライアントに対して送信し、

前記ファイルのうち、前記トークンによって関連づけられていない領域のデータを、前記ストレージに書き込むことによって、前記ストレージに対する同期処理を行うことを特徴とする請求項2又は3に記載の分散ファイルシステム。

【請求項5】

前記データ出力手段は、前記保持しているファイルを前記トークンを前記サーバに要求しているクライアントに対して送信するか、前記ストレージに書き込んで、前記ストレージに対する同期処理を行うかを、前記ネットワークと前記ストレージとの入出力性能及び／又は前記トークンを前記サーバに要求しているクライアントに対して送信されるファイルのデータ量に基づいて決定することを特徴とする請求項2から4のいずれか一つに記載の分散ファイルシステム。

【請求項6】

ファイルを保持するストレージと、前記ストレージに対してファイルの操作を行う複数のクライアントと、トークンによって前記クライアントによるファイルの書き込み及び読み出しの操作の権限を管理するサーバと、がネットワークによって接続されて構成される分散ファイルシステムに用いられるファイル送受信方法であって、

前記クライアントが、前記サーバに対して、ファイルの操作権限に関するトークンを要求し、

前記サーバは、前記ファイルの書込操作権限を保持しているクライアントに対して送信される、該書込操作権限に関するトークンの返却を要求するトークン返却要求に、該ファイルに関するトークンを要求しているクライアントの情報と、該クライアントが要求しているトークンの内容を示す情報とを含めて送信することを特徴とするファイル送受信方法。

【請求項7】

前記クライアントは、前記トークン返却要求を受信すると、該ファイルに関するトークンを要求しているクライアントに対して、前記記憶部に保持している該

トークンに関するファイルを送信することを特徴とする請求項 6 に記載のファイル送受信方法。

【請求項 8】

前記クライアントから前記トークンを前記サーバに要求しているクライアントに対して送信される該トークンに関するファイルは、前記ストレージに最新情報が反映されていないものであることを特徴とする請求項 7 に記載のファイル送受信方法。

【請求項 9】

前記トークンには、前記ファイルの領域に関連づけられており、

前記クライアントは、

前記ファイルのうち、前記トークンによって関連づけられた領域のデータを、前記トークンを前記サーバに要求しているクライアントに対して送信し、

前記ファイルのうち、前記トークンによって関連づけられていない領域のデータを、前記ストレージに書き込むことによって、前記ストレージに対する同期処理を行うことを特徴とする請求項 7 又は 8 に記載のファイル送受信方法。

【請求項 1 0】

前記クライアントは、前記保持しているファイルを前記トークンを前記サーバに要求しているクライアントに対して送信するか、前記ストレージに書き込んで、前記ストレージに対する同期処理を行うかを、前記ネットワークと前記ストレージとの入出力性能及び／又は前記トークンを前記サーバに要求しているクライアントに対して送信されるファイルのデータ量に基づいて決定することを特徴とする請求項 7 から 9 のいずれか一つに記載のファイル送受信方法。

【請求項 1 1】

ファイルを保持するストレージと、前記ストレージに対してファイルの操作を行う複数のクライアント装置と、前記クライアント装置によるファイルの書き込み及び読み出しの操作の権限をトークンによって管理するサーバと、がネットワークによって接続されて構成される分散ファイルシステムに用いられるクライアント装置であって、

前記ストレージから読み出したファイルのデータを保持する記憶部と、

前記サーバから送信された、前記ファイルの書込操作権限に関するトークンの返却の要求を受信すると、該ファイルに関するトークンを要求しているクライアント装置に対して、前記記憶部に保持している該トークンに関するファイルを送信するデータ出力手段と、を備えることを特徴とするクライアント装置。

【請求項 1 2】

前記トークンを前記サーバに要求しているクライアント装置に対して送信される該トークンに関するファイルは、前記ストレージに最新情報が反映されていないものであることを特徴とする請求項 1 1 に記載のクライアント装置。

【請求項 1 3】

前記データ出力手段は、

前記ファイルのうち、前記トークンによって関連づけられた領域のデータを、前記トークンを前記サーバに要求しているクライアントに対して送信し、

前記ファイルのうち、前記トークンによって関連づけられていない領域のデータを、前記ストレージに書き込むことによって、前記ストレージに対する同期処理を行うことを特徴とすることを請求項 1 1 又は 1 2 に記載のクライアント装置。

【請求項 1 4】

前記データ出力手段は、前記保持しているファイルを前記トークンを前記サーバに要求しているクライアントに対して送信するか、前記ストレージに書き込んで、前記ストレージに対する同期処理を行うかを、前記ネットワークと前記ストレージとの入出力性能及び／又は前記トークンを前記サーバに要求しているクライアントに対して送信されるファイルのデータ量に基づいて決定することを特徴とする請求項 1 1 から 1 3 のいずれか一つに記載のクライアント装置。

【請求項 1 5】

ストレージとネットワークによって接続されたクライアントによるファイルの書き込み及び読み出しの操作の権限のトークンによる管理をサーバ装置に実行させるプログラムであって、

前記サーバ装置を、前記ファイルの書込操作権限を保持しているクライアントに対して該書込操作権限に関するトークンの返却を要求するトークン返却要求を

送信するトークン返却要求手段として機能させるプログラム。

【請求項 16】

ネットワークによって接続されたストレージに記憶されたファイルの書き込み及び読み出しの操作の権限が、サーバが管理するトークンによって制御されるクライアント装置において実行されるプログラムであって、

前記サーバから送信された、前記ファイルの書込操作権限に関するトークンの返却の要求を受信すると、該ファイルに関するトークンを要求しているクライアント装置に対して、前記記憶部に保持している該トークンに関するファイルを送信する手段として機能させるプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、複数のクライアント（ノード）によってファイルの共有が可能なトークン方式の分散ファイルシステムにおいて、システムのスループットの向上に関するものである。

【0002】

【従来の技術】

従来の分散ファイルシステムは、トークンを利用してデータのコンシステンシ（一貫性）を保つのが知られている。このトークンには、データ、属性、サイズ、名前等の種類があり、それぞれのトークンについてリードとライトという状態（ステート）がある。例えば、データリードトークンを保持しているノードは、トークンに対応するファイルへの読み込みが許され、データライトトークンを保持しているノードは、トークンに対応するファイルへの書き込みが許される。また、リードトークンは複数のノードが同時に保持することができるが、ライトトークンはあるひとつのノードのみ保持できる。したがって、同一のファイルは複数のノードが同時に読み込むことができるが、複数のノードが同時に書き込むことができない。

【0003】

さらにデータトークンに関しては、ファイル全体に対して権限を与えるトーク

ンを発行する方式と、データの中のある部分（例えば、ファイルの先頭何バイト目から何バイト目）に限定した範囲に対する権限を与えるトークンを発行する方式がある。

【0004】

このトークンは、例えば、図5のようなフォーマットで構成されている。ファイルIDフィールド31は、どのファイルに対してなされたトークンであるかを示す。リボークトークン種フィールド32は、トークンの種類（データ、属性、サイズ、名前等）を示す。範囲指定フィールド33は、データの中の範囲を指定する。

【0005】

この従来のトークン方式の分散ファイルシステムでは、

- (1) 複数のノードが読み込み専用で同じファイルを参照する、
- (2) あるファイルに対してひとつのノードが読み込み、書き込みを行う、

ようなケースでは非常に効率的なファイルアクセスが行える。一方、

- (3) 複数のノードが同じファイルに対して読み込み、書き込みを行う、

ようなケースでは、パフォーマンスが極端に低下する。

【0006】

上記(3)のケースの従来の処理の流れを図6のフローチャートを参照して説明する。ここで前提条件として、クライアントAがファイルXのライトデータトークンを保持しており、ファイルXの最新更新データをディスクに未反映（書込前）の状態であるとし、前提条件における処理の流れを太線の矢印で示す。

【0007】

まず、クライアントBがファイルXに対してI/O処理（書込処理又は読出処理）を行った場合（処理101）、クライアントBはファイルXに対して適切なトークン（リード処理ではリードトークン、ライト処理ではライトトークン）を保持しているか否かを、自身のトークン保持テーブルを参照して判定する（処理102）。ファイルXに対する適切なトークンを保持していると判定された場合は処理110に移行する。

【0008】

一方、クライアント B はファイル X に対する適切なトークンを保持していない場合は、サーバ C に対して適切なトークンを要求する。サーバ C は、トークンの要求を受けると、ファイル X のライトトークンを保持しているクライアント（計算機）があるか否かを、自身のトークン管理テーブルを参照して判定する（処理 103）。ファイル X のライトトークンを保持しているクライアントがなければ処理 108 に移行する。一方、クライアント A がファイル X に対するライトトークンを保持しているため、サーバ C はクライアント A に対しトークンリボーク（返却）を要求する。

【0009】

クライアント A はサーバ C からのリボーク要求を受けると、自身のキャッシュ内に、ファイル X のデータであってディスクに未反映の（未だ書き込んでいない）ダーティーデータがあるか否かを判定する（処理 104）。ダーティーデータがない場合は処理 106 に移行する。一方、クライアント A にダーティーデータが存在するため、クライアント A はファイル X のダーティーデータをディスク（又はサーバ）に書き戻す（処理 105）。次に、クライアント A は自身のトークン保持テーブルからファイル X のライトトークンを削除し、サーバ C に対してトークンの返却を通知する（処理 106）。サーバ C はトークンの返却通知を受けると、トークンの返却を自身のトークン管理テーブルに反映する（処理 107）。次に、サーバ C は、クライアント B に対して要求トークンの付与をトークン管理テーブルに反映し（処理 108）、クライアント B に対し要求トークンを付与する旨のメッセージを送る。クライアント B はメッセージを受け取ると、自身のトークン保持テーブルにトークンの保持を反映する（処理 109）。クライアント B のトークンの反映が完了するとディスク（又はサーバ）に対して I/O 処理を行することができる（処理 110）。

【0010】

すなわち、あるノードがあるファイルに対してライトデータトークンを保持している状態で、同じファイルに対して別のノードがアクセスしたときに、4 回のノード間通信と、2 回のディスク I/O が発生していた。

【0011】

【特許文献 1】

特開 2000-322306 号公報

【非特許文献 1】

日本 SGI 株式会社、“CXFS：クラスタファイルシステム”、[online]、2001 年 11 月 5 日、インターネット<URL: <http://www.sgi.co.jp/products/pdf/sancxfswp.pdf>>

【0012】**【発明が解決しようとする課題】**

従来の分散ファイルシステムでは、あるノードがトークンを保有しているファイルに対して、他のノードが I/O 処理をする場合に、2 回のディスク I/O 処理と 4 回のノード間通信が発生し、しかもこの処理は順番に実行しなければならないので、ディスクの I/O 処理に時間を必要とし、システムの性能が低下していた。

【0013】

本発明は、上記の不具合を鑑みてなされたものであり、ディスクの I/O 処理を減らした分散ファイルシステムを提供することを目的とする。

【0014】**【課題を解決するための手段】**

本発明は、サーバがトークンを保持しているノードに対してリボーク要求を発行する際に、トークンを要求しているノードの情報を付加したりボーク要求を発行することにより、同じファイルをストレージに書き戻さず、ダーティーな状態のままのファイルを渡すので、ストレージへの I/O 処理回数を可能な限り減らし、処理を並列化する。

【0015】**【発明の作用と効果】**

本発明は、あるノードがあるファイルに対してライトデータトークンを保持している状態で、同じファイルに対して別のノードがアクセスしたときに、ファイルをストレージに書き戻さず、ダーティーな状態のままのファイルを渡すので、ストレージへのアクセス回数を減らすことができ、理想的なケースでは 4 回のノ

ード間通信のみ（ディスク I/O なし）でファイルへのアクセスが可能となり、システム全体としてのスループットを向上することができる。

【0016】

例えば、ノード間で共有しているログファイルの処理では、複数のノード間で短いデータの追記書きが発生するため、ディスクの I/O 回数を軽減させ、スループット性能の性能向上が期待できる。

【0017】

【発明の実施の形態】

以下に、本発明の実施の形態を図面を参照して説明する。

【0018】

図1は、本発明の第1の実施の形態の分散ファイルシステムの構成を表したブロック図である。

【0019】

クライアント10、11は分散ファイルシステムのノードとして機能する計算機である。このクライアント10、11とストレージシステム16とがLAN14によって接続されている。

【0020】

クライアント10、11の各々には、トークン保持テーブル10b、11b、キャッシュ10c、11cが備えられている。トークン保持テーブル10b、11bは、クライアントが保持しているトークンの内容を記憶する。キャッシュ10c、11cは、ストレージシステム16から読み込んだデータを一時的に保管するためのメモリ装置である。このキャッシュ10c、11cには、キャッシュに保持されているデータがダーティーである（ストレージ装置に未だ書き込まれていない未反映のデータがある）か否かを示すダーティーフラグが備えられている。

【0021】

ストレージシステム16には、サーバ12、ストレージ装置13が備えられており、Network Attached Storage (NAS) として機能している。サーバ12とストレージ装置13とはStorage Area Network (SAN) 15によって接続され

ている。このストレージ装置 13 は、ハードディスク等の記憶装置で構成されている。よって、クライアント 10、11 は、LAN 14 を経由してストレージシステム 16 に備えられたストレージ装置 13 に記憶されたデータの読み出し及び書き込みができる。

【0022】

サーバ 12 には、トークン管理テーブル 12a、トークン保持テーブル 12b、キャッシュ 12c が備えられている。トークン管理テーブル 12a は、全てのクライアントが、それぞれどのような内容（例えばリード、ライト等）のトークンを保持しているかを該当するデータに対応付けて記憶する。トークン保持テーブル 12b は、サーバが保持しているトークンの内容を記録する。キャッシュ 12c は、データを一時的に保管するためのメモリである。

【0023】

次に、以上のように構成された本発明の第 1 の実施の形態の動作について説明する。

【0024】

各クライアント 10、11 はサーバ 12 からトークンを付与されることによってファイルへのアクセスが許され、トークンを管理しているサーバ 12 は、トークン管理テーブル 12c を参照することによって、現在、どのクライアントによって、どのファイルに対してどのような処理が行われているのかを知ることができる。

【0025】

例えば、クライアント 10 にファイル A に対するライトデータトークンが付与されると、クライアント 10 は、ストレージ装置 13 からファイル A を読み込んだり、ファイル A を書き込みモードでキャッシュ 10c に保持することが可能になる。続いて、クライアント 11 がファイル A に対するリードデータトークンを要求した場合、サーバ 12 が、クライアント 11 からのリードデータトークン要求を受け取ると、トークン管理テーブル 12c を参照することで、ファイル A がクライアント 10 が保有するライトデータトークンによって書き込みモードで保持されていることがわかる。そのため、サーバ 12 はクライアント 10 に対して

トークンのリボーク（返却）要求を行う。リボーク処理が完了すると、クライアント 11 にファイル A に対するリードデータトークンが付与され、クライアント 11 によるファイル A の読み込みが可能となる。

【0026】

図 2 は、第 1 の実施の形態のクライアント 10、11 とサーバ 12 のファイルの I/O 処理（ファイルの読み込み、書き込み等）のフローチャートを示す。

【0027】

なお、このフローチャートの処理の前提条件として、クライアント 10 がファイル X に対するライトデータトークンを保持しており、ファイル X の最新更新データがディスクに未反映の状態、すなわちファイル X にダーティーデータを含んでいる状態を前提し、前提条件における処理の流れを太線の矢印で示す。

【0028】

まず、クライアント 11 において、ファイル X を読み込む必要が生じ、分散ファイルシステム上のファイル X に対する I/O 要求を発行する（処理 201）。

【0029】

クライアント 11 は、この I/O 要求を受けて、まず、自身のトークン保持テーブル 11b がファイル X の適切なトークンを保持しているか否かを判定する（処理 202）。適切なトークンを保持している、すなわち、ファイル X に対する読み込み処理ならばリードデータトークン、ファイルに対する書き込み処理ならばライトデータトークンを保持していると判定した場合には、サーバ 12 に対して I/O 処理を実行し、ストレージ装置 13 よりファイルを読み込む（処理 218）。

【0030】

処理 202 において適切なトークンを保持していないと判定した場合には、サーバ 12 に対してトークンを要求するトークン要求メッセージを送る。サーバ 12 はトークン要求メッセージを受け取ると、ファイル X のライトデータトークンを保持するクライアントがトークン管理テーブル 12a に登録されているか否かを判定する（処理 203）。ライトデータトークンを保持しているクライアントが無い場合は、直ちにトークンを付与することが可能であるので、トークン管理

テーブル 12 a にファイル X に対するクライアント 11 のライトデータトークンの付与を反映し（処理 216）、クライアント 11 にライトデータトークンを付与した旨を通知する。

【0031】

クライアント 11 は通知を受け取ると、トークン保持テーブル 11 b に、処理 202 で要求したトークンの種類（ライト／リード等）を登録し（処理 217）、サーバ 12 に対して I/O 処理を実行し、ストレージ装置 13 よりファイルを読み込む（処理 218）。

【0032】

処理 203 において、ライトデータトークンを保持しているクライアントが既に存在すると判定した場合には、ライトデータトークンを保持しているクライアント（クライアント 10）に対してリボーク要求メッセージを送る（処理 204）。このリボーク要求メッセージには、図 3 にて後述するように、トークンを要求しているノード（クライアント 11）、要求しているトークンのステート（クライアント 11 が要求しているステート（リード又はライト））の情報が含まれている。

【0033】

クライアント 10 は、リボーク要求メッセージを受け取ると、ライトデータトークンのリボーク処理（ライトデータトークンの返却とファイルの受け渡し）を行う。まず、トークン保持テーブル 10 b からファイル X に対するライトデータトークンを削除する（処理 205）。次に、クライアント 10 のキャッシュ 10 c にあるファイル X のデータがサーバ 12 に反映されているかどうか、すなわちファイル X にダーティーデータが存在するか否かを判定する（処理 206）。ダーティーデータが存在すると判定した場合には、ダーティーデータを含んだファイル X をトークン要求元（クライアント 11）に対して送信する（処理 207）。このトークン要求元はトークン要求メッセージに含まれているトークン要求ノードより得る。ファイル X を送ると処理 208 に移行する。

【0034】

一方、処理 206 においてダーティーデータが存在しないと判定した場合には

、処理 208 に移行する。

【0035】

次に、ファイル X を受け取ったトークン要求元（クライアント 11）は、受信したデータを自身のキャッシュ 11c に取り込む（処理 211）。このとき、クライアント 11 が処理 202 において要求したトークンの種類がリードデータトークンである場合には（処理 212）、直ちにファイル X をサーバ 12 に書き出す処理（ファイルの同期処理）を行う（処理 213）。これは、リードデータトークンのみを保有しているクライアントがダーティーデータを含むファイルを持つと、ファイルの一貫性（コンシステンシ）が失われるためである。すなわち、複数のクライアントが同時にファイルを読み出すことができるので、ディスクに未反映のデータがあるとファイルの一貫性が保てないためである。

【0036】

このとき、クライアント 10 は、処理 205 で受け取ったリボーク要求メッセージからトークン要求元の要求トークンステートフィールドを参照して、要求トークンの種類（要求トークンがリードであるかライトであるか）を判定する（処理 208）。トークン要求元がライトデータトークンを要求していると判定した場合には、自身の持つキャッシュ 10c にあるファイル X のダーティーデータを全て破棄する（処理 209）。これは、同一のファイルに対して複数のクライアントにライトトークンを付与することはできないため、他のノードがライトデータトークンを要求した場合には、リボーク処理によってファイルの所有権を新たにライトデータトークンを保有するクライアントに完全に移行するためである。処理が完了すると、トークン要求元（クライアント 11）に対して自身のトークンステートの変化を通知する。

【0037】

一方、処理 208 でトークン要求元がリードデータトークンを要求していると判定した場合には、キャッシュ内にあるファイル X のダーティーデータに対応するダーティー状態を示すダーティーフラグをクリアし、データ自体は破棄せず読み込み専用データに変更してキャッシュ 10c に保存しておく。次に、自身のトークン保持テーブルにファイル X のリードデータトークンを登録する（処理 21

0)。これは同一のファイルに対してリードデータトークンは複数のクライアントが持つことができるためであり、ファイルXの中のダーティーデータは処理213によってディスクに反映されるからである。処理が完了すると、トークン要求元（クライアント11）に対して自身のトークンステートの変化を通知する。

【0038】

ここで、クライアント11は、処理209又は処理210によって、クライアント10からトークンステートの変化の通知を受け、さらに処理213が完了すると、クライアント10（トークンリボーク要求先）と自身（トークン要求元であるクライアント11）のトークンステートの変化をサーバ12に送信する（処理214）。なお、このクライアント10のトークンステートの変化を、クライアント11を経由することなく、クライアント10からサーバ12に直接送信するようにしてもよい。

【0039】

サーバ12はこの変化をトークン管理テーブル12aに反映する（処理215）。

【0040】

次に、クライアント11は、自身のトークン保持テーブル11cに、処理202で要求したトークンの登録を行う（処理217）。そして、必要ならばサーバ12に対してI/O処理を実行し、ストレージ装置13よりファイルを読み込む（処理218）。

【0041】

以上の処理によって、クライアント10に対するトークンのリボーク処理が完了し、ダーティーデータを含んだデータXがクライアント10からクライアント11へ渡される。

【0042】

なお、上記の処理では、クライアント10からクライアント11に対してダーティーデータを含んだファイルXの送信（処理207→処理211）とトークン状態変化通知の送信（処理209、210→処理214）は別に行っているが、ひとつのメッセージにまとめて一度に送信することも可能である。

【0043】

また、図2の処理207において、クライアント11に送る代わりにストレージ装置13に書き出した方が性能上有利である場合、すなわちクライアント（またはサーバ12）が、LANのスループット性能やストレージ装置のI/Oのスループット性能等から予め設定しておいた性能予測式に基づいて、ディスクのI/Oスループット性能がLANのスループット性能より高いと判断した場合には、要求クライアントに対してファイルを送らずに、ストレージ装置13に書き出す処理を行いダーティー状態でなくなったファイルを、要求クライアントが読み出すようにしてもよい。こうすることで、スループットの高い処理手段を優先でき、分散ファイルシステム全体としてのスループットが向上する。

【0044】

図3に、図2に処理204において送信されるトークン要求メッセージの詳細を示す。

【0045】

トークン要求メッセージは、ファイルIDフィールド41、リボークトークン種フィールド42、範囲指定フィールド43、要求トークンステートフィールド44、トークン要求ノードIDフィールド45から構成される。

【0046】

ファイルIDフィールド41は、トークンの要求を行うファイルの情報を示している。リボークトークン種フィールド42は、トークンの種別（例えばデータ、属性、サイズ、ファイル名等）を示す。範囲指定フィールド43は、ファイルの範囲を示す。これはトークンによってファイルの範囲指定が可能な分散ファイルシステムにおいて、この範囲指定フィールド43で指定した範囲内のみでクライアントが処理を行うことができるものである。要求トークンステートフィールド44は、トークンのステート（リード／ライト）を示す。トークン要求ノードIDフィールド45は、トークンを要求している要求元のクライアントの情報を示す。

【0047】

以上のように構成された本発明の第1の実施の形態の分散ファイルシステムで

は、あるクライアントが持っているダーティーデータを含んだファイルを他のクライアントが要求した場合に、ファイルをストレージに書き戻さず、ダーティーな状態のままのファイルを他のクライアントに渡すので、ストレージへのアクセス回数を減らすことができ、分散ファイルシステム全体のスループットを向上することができる。

【 0 0 4 8 】

次に、本発明の第 2 の実施の形態の分散ファイルシステムについて説明する。第 2 の実施の形態では、第 1 の実施の形態と比較すると、各クライアントが直接ストレージと S A N によって接続されており、サーバを経由することなくストレージ上のデータを読み書きできる点が相違する。なお、第 1 の実施例と同一の動作をする構成には同一の符号を付し、その説明は省略する。

【 0 0 4 9 】

図 4 は第 2 の実施の形態の分散ファイルシステムの構成を表したブロック図である。

【 0 0 5 0 】

クライアント 1 0、1 1 とストレージシステム 2 6（サーバ 2 2）とが L A N 2 4 によって接続されている。

【 0 0 5 1 】

ストレージシステム 2 6 には、サーバ 2 2、ストレージ装置 1 3 が備えられており、サーバ 2 2 とストレージ装置 1 3 とは S A N 2 5 によって接続されている。

【 0 0 5 2 】

また、クライアント 1 0、1 1 も、ストレージ装置 1 3 と S A N 2 5 によって接続されている。そのため、クライアント 1 0、1 1 はサーバ 2 2 を介することなくストレージ装置 1 3 にあるファイルを読み込み、ストレージ装置 2 3 に対してファイルを書き込むことができる。すなわち、クライアント 1 0、1 1、サーバ 2 2、ストレージ装置 1 3 が S A N を構成している。

【 0 0 5 3 】

サーバ 2 2 には、トークン管理テーブル 2 2 a、トークン保持テーブル 2 2 b

、キャッシュ 22c が備えられている。トークン管理テーブル 22a は、各クライアントから発行されたトークンを受け取り、受け取ったトークンは、どのクライアントから、どのような内容（例えばリード、ライト等）のトークンであるかを該当するデータに対応付けて記憶する。トークン保持テーブル 22b は、クライアントが要求したトークンの内容を保持する。キャッシュ 22c は、データを一時的に保管するためのメモリである。

【0054】

次に、上記のように構成された第 2 の実施の形態の分散ファイルシステムについて、次に動作を説明する。

【0055】

本実施の形態の処理の流れは、第 1 の実施の形態（図 2）と基本的には同一であるが、クライアント 10、11 は、ストレージ装置 13 に対する I/O をサーバを介することなく行うことができる。図 2 の処理 213 では、ファイル X をサーバ 12 に対して送った後にストレージ装置 13 に書き込まれるが、第 2 の実施の形態では、クライアント 11 が直接ストレージ装置 13 に対して I/O を行いファイル X を書き込む。同様に、処理 218 では、クライアント 11 が直接ストレージ装置 13 に対してファイル X に対する I/O を発行する。

【0056】

以上のように構成された第 2 の実施の形態では、第 1 の実施の形態の効果に加え、クライアントがサーバ 12 を介することなくストレージ装置に I/O が行えるので、サーバ 12 の処理が軽減され、結果として分散ストレージシステム全体としてのスループットを向上することができる。

【図面の簡単な説明】

【図 1】 本発明の第 1 の実施の形態の分散ファイルシステムの構成を示すブロック図である。

【図 2】 同じく分散ファイルシステムの処理を示すフローチャートである。

【図 3】 同じくリボーク要求トークンの内容を示すフォーマット図である。

【図 4】 本発明の第 2 の実施形態の分散ファイルのシステムの構成を示すブロック図である。

【図5】 従来のトークンリボーク要求の内容を示すフォーマット図である。

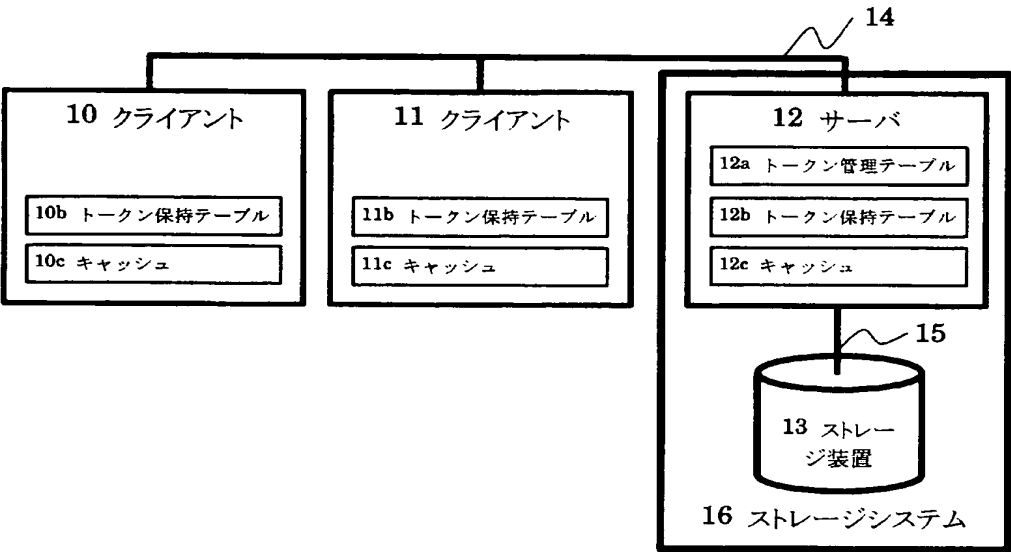
【図6】 従来の分散ファイルシステムの処理を示すフローチャートである。

【符号の説明】

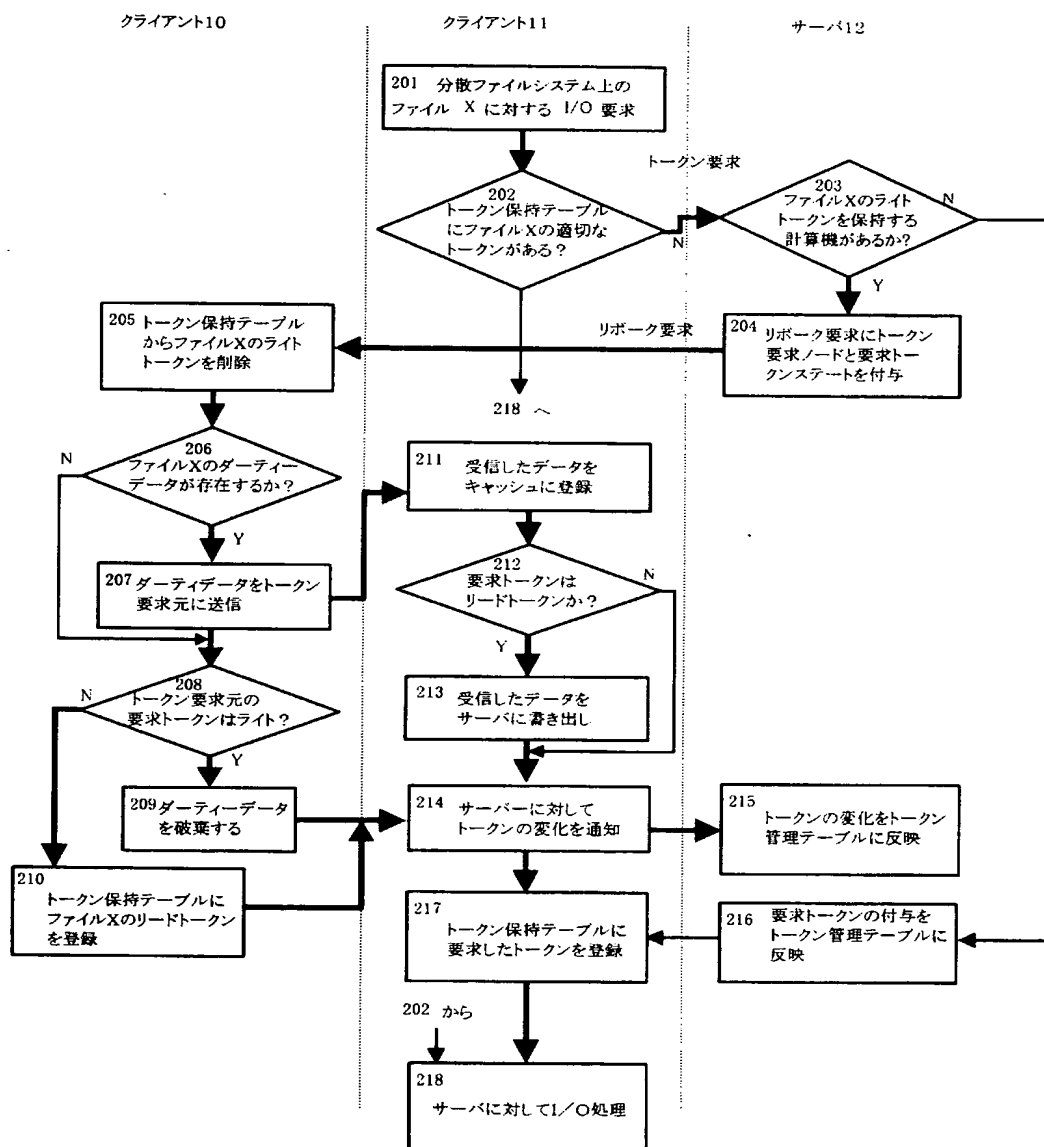
- 10、11 クライアント
- 10b、11b トークン保持テーブル
- 10c、11c キャッシュ
- 12 サーバ
- 12a トークン管理テーブル
- 12b トークン保持テーブル
- 12c キャッシュ
- 13 ストレージ装置
- 14 LAN
- 15 SAN
- 16 ストレージシステム
- 20 クライアント
- 22 サーバ
- 22a トークン管理テーブル
- 22b トークン保持テーブル
- 22c キャッシュ
- 23 ストレージ装置
- 25 SAN
- 26 ストレージシステム

【書類名】 図面

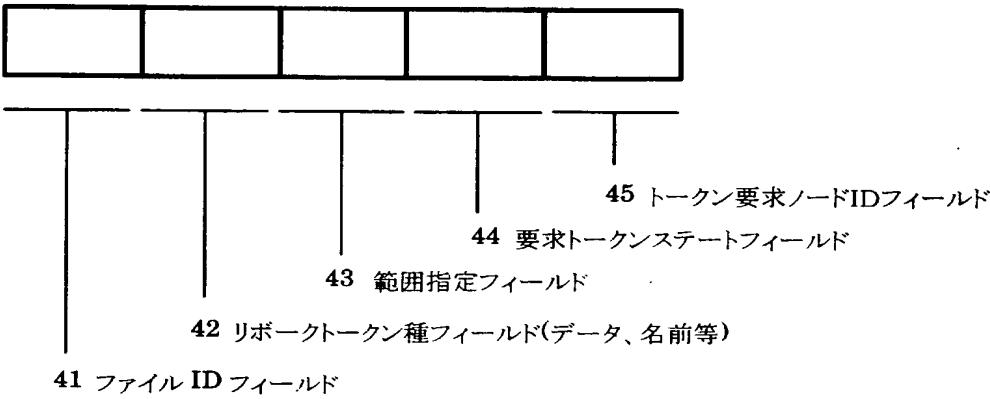
【図 1】



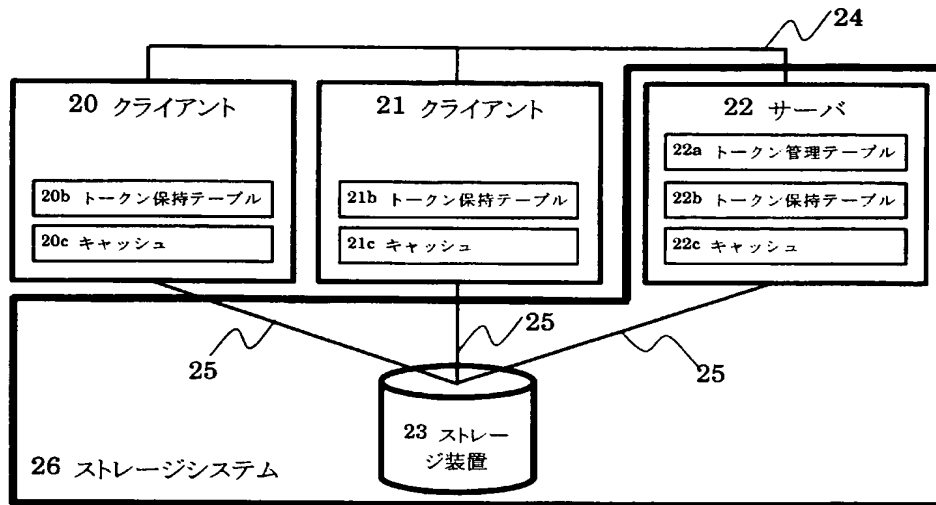
【図 2】



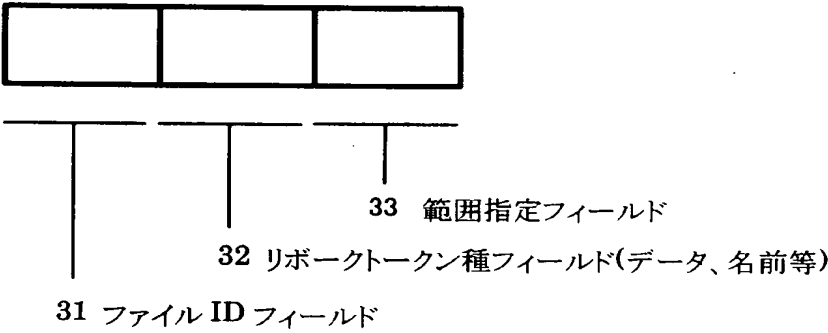
【図 3】



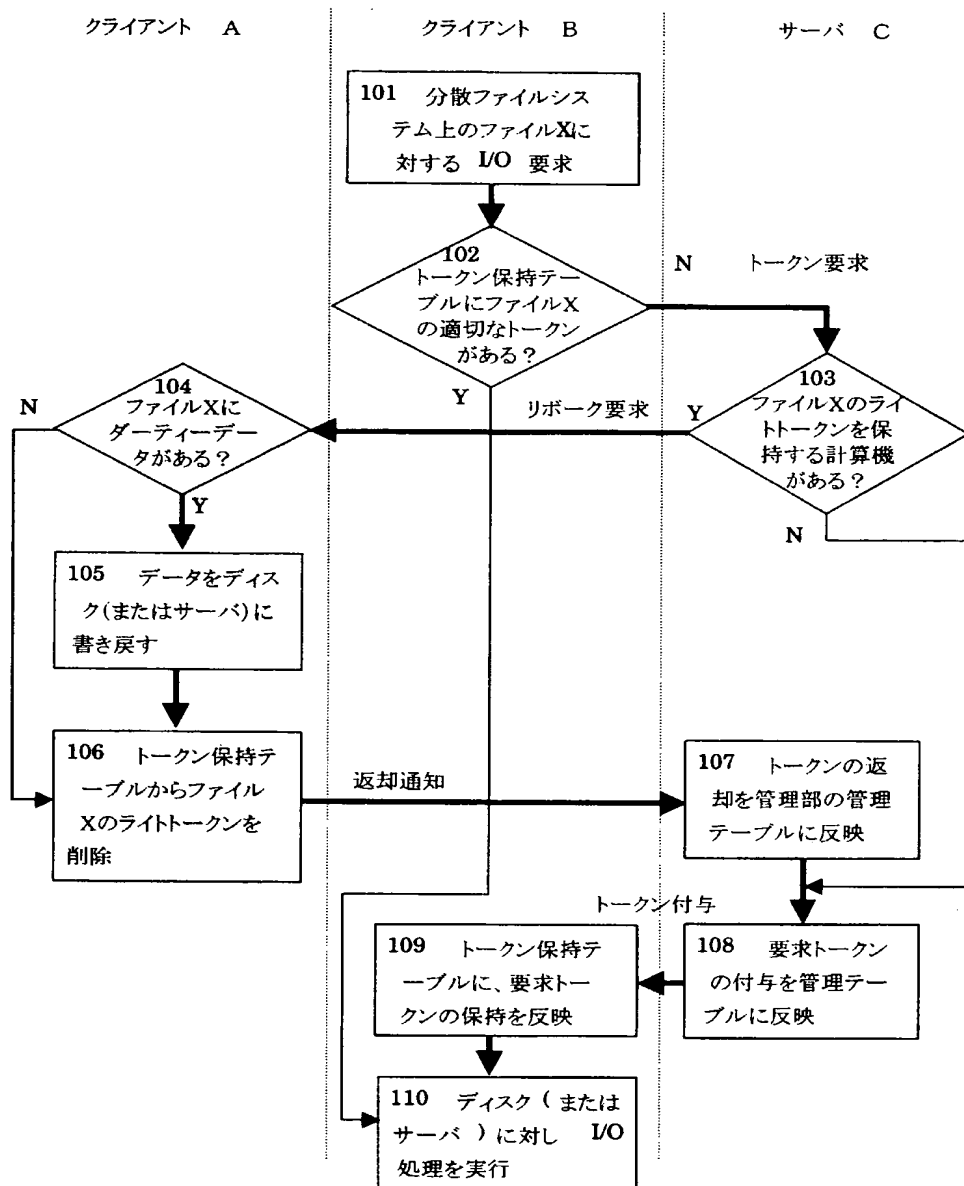
【図 4】



【図 5】



【図 6】



【書類名】 要約書**【要約】****【課題】**

複数のクライアント（ノード）によってファイルの共有が可能なトークン方式の分散ファイルシステムにおいて、システムのスループットの向上に関する

【解決手段】

ファイルを保持するストレージ 1 3 と、トークンの管理を行うサーバ 1 2 と、ストレージ 1 3 に対してファイルの読み書きを行う複数のクライアント 1 0 と、がネットワークで接続されており、前記クライアントがトークンによってファイルの要求を行う分散ファイルシステムにおいて、サーバ 1 2 は、ファイルを保持しているクライアント 1 0 に対してトークンの返却を要求するトークン返却要求手段を備え、前記トークン返却要求手段は、トークンによってファイルを要求しているクライアント 1 0 の情報と、トークンの内容を示す情報とを含めたトークン返却要求を送信する。

【選択図】 図 1

特願 2 0 0 3 - 0 6 3 5 6 9

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 1 0 8]

1. 変更年月日

1 9 9 0 年 8 月 3 1 日

[変更理由]

新規登録

住 所

東京都千代田区神田駿河台 4 丁目 6 番地

氏 名

株式会社日立製作所